❒ 140

From CAD to Robot: Undergraduate Capstone Design in Engineering Technology

Kuldeep S. Rawat[1], G.H. Massiha[2]
[1]Department of Technology, Elizabeth City State University, Elizabeth City, North Carolina 27909
[2]Department of Industrial Technology, University of Louisiana at Lafayette, Lafayette, Louisiana 70504

Article Info	ABSTRACT

A novel senior project in designing and implementing a wheeled platform-based experimental mobile robot is discussed. This mobile robot design project was used as a platform to learn sensor interfacing, microcontroller programming, motor control, and electronic circuit design and troubleshooting. A specially designed proto board was used so that students could experiment with various types of sensors and supporting electronic circuitry. The modules implemented in this project are, servo motor control, infrared (IR)-based obstacle detection and avoidance, temperature sensing, and IR wireless communication. An 8-bit Peripheral Interface Controller (PIC) microcontroller, operating at 20MHz, was used as a programmable controller to monitor external environment through sensors and make appropriate decisions. PIC microcontroller was programmed using PICBasic PRO, a BASIC like high-level language. The implementation was divided into separate experiments, through which the students progressively completed the mobile robot. This progressive experimentation helped students develop their knowledge of interfacing, microcontroller programming, electronic control, circuit design, and troubleshooting in an incremental manner. The robot design experiments, sensor interfacing, electronic control, supporting circuitry, problems faced and troubleshooting during implementation are discussed in the paper.

Corresponding Author:

Kuldeep S. Rawat
Department of Technology,
Elizabeth City State University
Elizabeth City, NC 27909
Email: ksrawat@mail.ecsu.edu

1. INTRODUCTION

An undergraduate senior design project course is an important part of an undergraduate Engineering Technology degree program, ABET, 2011 [1]. The senior design course gives students the opportunity to demonstrate what they have learned throughout their studies. Students apply basic theories to experimental projects related to real-world problems. These projects aid the student in developing critical thinking, reasoning, and analytical skills that are crucial in finding solutions to design problems [1]. One such project done by a group of 3 undergraduate seniors at ECSU was the design and implementation of an autonomous experimental mobile robot. Each module was implemented and tested independently and then integrated to form a fully functional mobile robot.

An autonomous mobile robot can be used for a number of applications ranging from automated guided vehicles (AGV) in a warehouse to defusing bombs in urban combat [2]. Mobile robots are used by the military and SWAT teams to venture into hazardous locations to detect and defuse explosive devices, lay cabling, or open doors during an operation. Another application for autonomous mobile robots is fire fighting, where a team of three mobile robots work together in a fire fighting operation [3].

The mobile robot implemented in this project can be used for remote temperature sensing from a hazardous environment. The temperature data can be sent via infrared (IR) wireless communication to a host PC at a remote location. The project group first conceived 3-D model of the mobile robot in Pro E software as shown in Figure 1. This was followed by machining by students of the mobile robot base and wheel assembly. Once the base and wheels were manufactured and assembled the electronic control and sensor interface modules were added to the design.
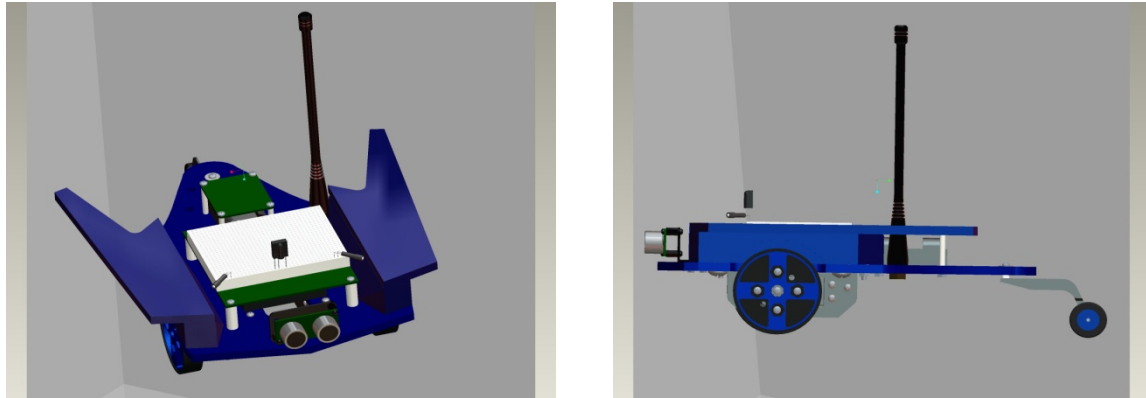
Figure 1. 3-D Model of Mobile Robot

The mobile robot designed and implemented in this senior project course has several features, such as obstacle avoidance and navigation, temperature sensing, and data communication using short range infrared (IR) technology. This mobile robot is an excellent learning platform to gain knowledge of sensor interfacing, microcontroller programming, motor control, and electrical circuitry design and testing. The remaining paper is organized as follows.

The next section discusses the hardware and software used in implementation of the project. Section 3 discusses the robot design experiments. In this section, the individual experiments/modules will be discussed in detail. Section 4 highlights the technical problems faced during implementation and troubleshooting. Finally, the relevant conclusion is presented.

2.   HARDWARE AND SOFTWARE

The basic block diagram for an intelligent mobile robot is shown in Figure 2. An intelligent mobile robot requires programmable hardware to act as its brain for decision making. Microcontrollers are the most commonly used programmable controllers for this purpose. It monitors sensory feedback or input from sensors and uses the information for making decisions regarding navigation, object detection and avoidance, and even communication [3].
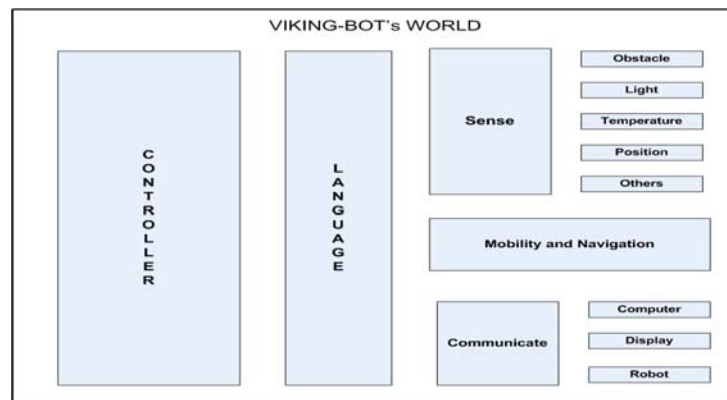
Figure 2. Basic block diagram of mobile robot

The microcontroller used in this design is the PIC16F628A, an 18-pin, flash-based, 8-bit CMOS microcontroller manufactured by Microchip in 2008 [4]. The PIC16F628A is characterized by low power consumption (about 0.6mA at 4MHz) and the integrated 2Kwords of flash program memory. This allows modification to the robot control program within minutes depending on particular application or new extension hardware. The PIC16F628A has two I/O ports, namely PORTA (5 bits) and PORTB (8 bits). Together the two ports provide 13 I/O lines. It also features 128 bytes of EEPROM data memory, a Capture/Compare/PWM (CCP) module, a Universal Synchronous-Asynchronous Receiver Transmitter (USART), 2 Comparators and a programmable Voltage Reference that make it ideal for analog/integrated level applications in automotive, industrial appliances, and other consumer applications [4].

MicroEngineering Labs (meLabs) PICBasic PRO are software used [5]. The PICBasic PRO compiler is specifically designed for programming PIC microcontrollers using Beginner's All-purpose Symbolic Instruction Code (BASIC) like high level language. The compiler comes with MicroCode Studio, a visual integrated development environment (IDE) for editing, running, and debugging the program code. A single PICBasic PRO command is enough to replace several lines of assembly language instructions. This makes code easier to debug and achieve significant reduction in programming time. The MicroCode Studio also has a Serial Communicator interface for the application requiring communicating with a host PC. The program code was burned onto the PIC microcontroller using meLabs USB programmer [4].

3.   MOBILE ROBOT DESIGN EXPERIMENTS
The mobile robot was designed, implemented, and tested in steps. Each module was implemented and tested independently and then integrated to form a fully functional mobile robot. The modules used in mobile robot are, servo motor control, IR obstacle detection and avoidance, Ultrasonic distance measurement, temperature sensing, and IR wireless communication. These modules/experiments are discussed next.

3.1. Controlling Servo Motors Using PIC16F628A
To provide the maneuvering ability (mobility) around obstacles in its path, the mobile robot uses two servo motors. Servo motors used are the Grand Wing (GW) standard servo motors with a torque of 47-56oz-in and an operating voltage between 4.8-6V. Each servo motor is pre-modified for continuous rotation and comes with Fatuba spline control horn. The wheels used were machined out of lightweight but sturdy expanded poly vinyl chloride (PVC) material. The wheels weigh 26gms, measure 2½" in diameter, with tread (made from polypropylene rubber) width of ½" and are attached to the servo horns.

The servos are controlled by Pulse Width Modulation (PWM) signals sent through the command/control wire. The angular position is determined by the duration of a pulse that is applied to the control wire. The standard servo timing pulses are shown in Figure 3.
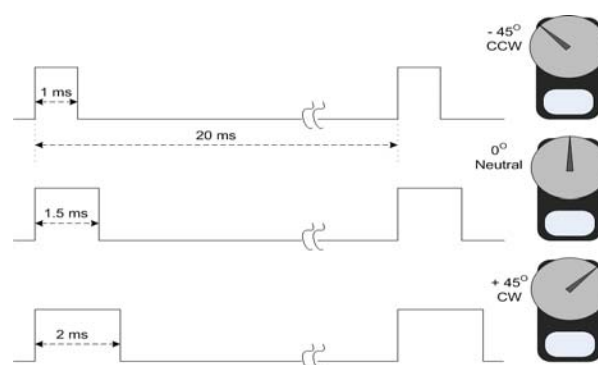

Figure 3. Servo Motor Pulses

The servo expects to see a pulse every 20ms or approximately 50 times a second (50 Hz). The servo control pulses were generated using PICBasic PRO PULSOUT command. The PULSOUT command has a resolution or period of 2µs when operating at 20MHz. Even though, the mobile robot uses servo motors that have been pre-modified for continuous rotation, they still require the same position pulses for rotation. Referring to Figure 3, a pulse of 1ms (1000 µs) or servo direction constant of 500 (500 x 2µs = 1000µs) will

cause counter clockwise (CCW) rotation of the motor, while a pulse of 2ms or direction constant of 1000 (1000 x 2μs = 2000μs) would cause clockwise (CW) motion. A pulse of 1.5ms or direction constant of 750 (750 x 2μs = 1500μs) is required to stop the servo motor. To have mobile robot travel in a straight line, at full load, slight adjustments is necessary to stop mobile robot from veering from his intended course. Students experimented with servo motors to determine the exact motor direction constants. The experimental direction constants for the two (left and right) servos are summarized in Table 1.

Table 1. Servo Motor Direction Constant

Directions	Left Motor	Right Motor
Slow CW	775	820
Fast CW	835	890
Stop	770	805
Slow CCW	760	795
Fast CCW	685	725

Once the exact direction constants were determined, the two servos were tested for forward, reverse, left, and right movements of mobile robot. The schematic with the two servo motors that drive the left and right wheels of the mobile robot controlled by PIC16F628A is shown in Figure 4.
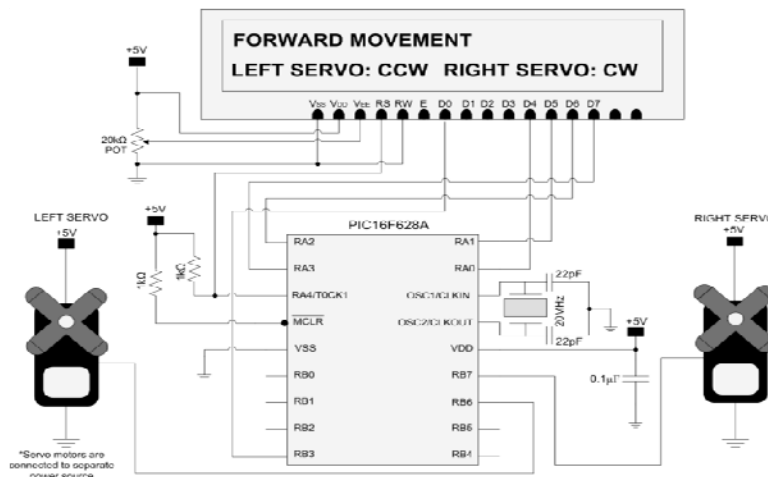

Figure 4. Two Servo Motors (Left and Right) Control Interface

A program was written to test the robot's mobility after both the left and right servo motors and wheels were mounted on the mobile robot's platform base. An LCD interface was used to display the direction of each servo motor as well as the robot. The utility program aided in testing servo operation, microcontroller functionality, regulator circuit, electrical connections, and batteries.

3.2. Infrared (IR) Obstacle Detection and Avoidance

A mobile robot requires an obstacle detection mechanism to aid in its navigation around the obstacles in its path [6], [7]. In this experiment, the obstacle detection and avoidance circuitry was set up using two LUMEX high power IR Light Emitting Diodes (LEDs) with wavelength of 940nm and a TSOP1740 IR detector module with band-pass frequency of 40kHz. The obstacle detection interface is shown in Figure 5. An object in the path of the IR LED's beam reflects the IR energy back onto the face of the infrared detector module. The IR LED is modulated (switched on and off) by a 40kHz PWM carrier signal generated by PIC16F628A microcontroller. When no infrared energy, modulated at 40kHz is detected (idle), the output state of the infrared detector module is logic 1 or +5V, and when IR energy modulated at 40kHz is striking the face of the detector the state of IR detector's output is logic 0. The internal hardware PWM feature of the PIC16F628A is used to generate the 40kHz PWM signal (Microchip, 2008) [4]. The 40kHz PWM signal is applied to the anode side of each infrared LED, as shown in Figure 5.
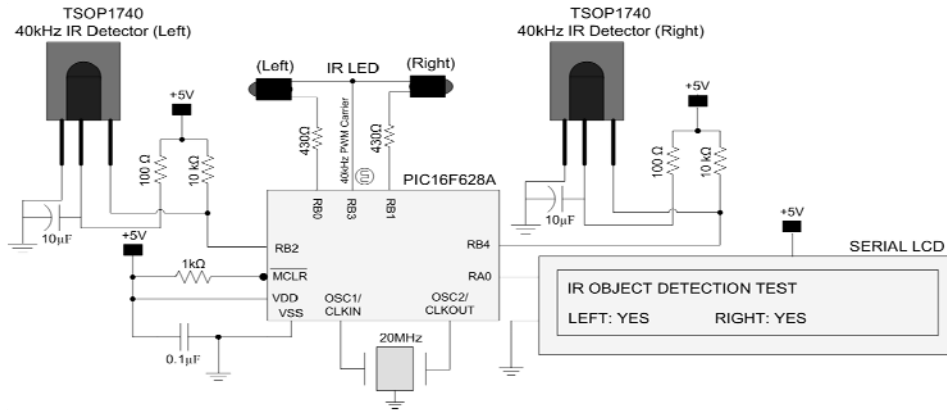
Figure 5. IR Obstacle Detection Interface

The IR obstacle detection circuit has PORTB I/O-pins (PORTB.0 and PORTB.1) each connected to IR LED cathode side to control each IR LED, as seen in Figure 5. When the 40kHz PWM signal is on, logic 0 (ground) on PORTB.0 will turn on the LED. Logic 1 (HIGH) on PORTB.0 will turn off the LED. The outputs of IR detector are sampled at PORTB.2 and PORTB.4 pins, once the IR LED cathode is made low. A logic 0 indicates the IR energy is striking the detector, and an object is in the robot's path. A logic 1 on the detector's output indicates there is no object present to reflect IR energy back onto the face of the detector. Once the circuit was set up, the PIC16F628A was programmed to control the IR LEDs and read IR detector outputs. The circuitry was tested by placing objects at different angle in front of IR LEDs. For the experimental purpose a serial LCD was connected to display the presence or absence of an object.

3.3. Distance Measurement Using Ultrasonic RangeFinder

An ultrasonic sensor, like radar, uses the principle of echo location. An echo signal is created when a short pulse is sent in a specific direction and hits an object that does not absorb the pulse [8]. This echo can be picked up by a detector circuit. By measuring the time between sending the pulse and detecting the echo, the distance to the object can be determined.

The ultrasonic rangefinder used in this project is a Devantech SRF04 module. The Devantech SRF04 ultrasonic rangefinder provides precise, non-contact distance measurements from about 3 cm (1.2 inches) and 3 meters (3.3 yards). The sonar module interface diagram is shown in Figure 6. It only requires two I/O pins to connect to the PIC microcontroller. Output from the SRF04 is in the form of a variable-width pulse that is proportional to the distance to the target. The timing pulses required for the operation of SRF04 module are shown in Figure 7.
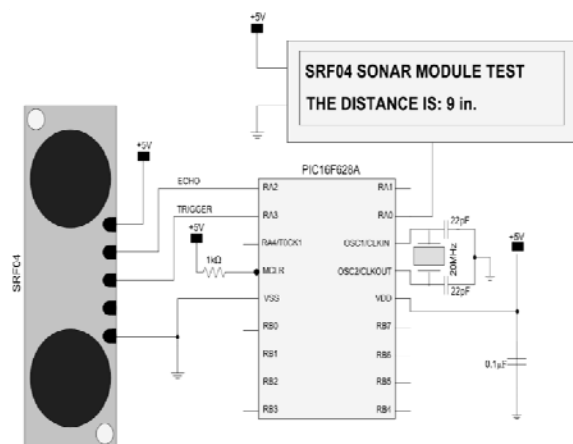

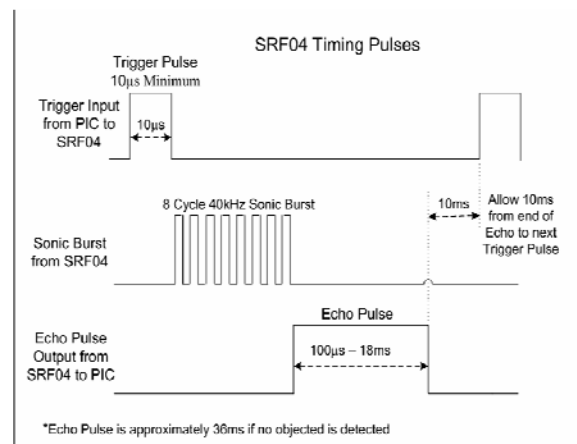Figure 6. SRF04 Ultrasonic Module Interface

Figure 7. SRF04 Timing Pulse

A short 10μs pulse is sent to the TRIGGER input of the SRF04 by the PIC16F628A to start the ranging. The TRIGGER input is connected to PORTA.3 of the PIC microcontroller. The SRF04 sends out an 8 cycle burst of ultrasound at 40kHz that travels through the air at speed of about 1.125 ft/ms and raises its ECHO line high. The ECHO line is connected to PORTA 2 of the PIC microcontroller. It then listens for an echo and as soon as it detects one it lowers the ECHO line again. The ECHO line is therefore a pulse whose width is proportional to the distance to the object.

By timing the pulse it is possible to calculate the range in inches or centimeters. If no object is detected then the SRF04 will lower its ECHO line after 36ms. The trigger pulse must be at least 10μs long. Again, PICBasic PRO's PULSOUT command is used to generate this pulse. The other requirement is that there is at least 10ms of wait time between measurements. The PICBasic PRO's PULSIN command is used to measure the width of the pulse output on the SRF04 ECHO pin. The resolution for PULSIN is the same as with the PULSOUT command (2μs at 20MHz). The SRF04 outputs a pulse on the Echo pin after each ultrasonic measurement. The resulting pulse width returned by the PULSIN command is divided by a factor of 74 to determine the distance in inches. After the PIC microcontroller was programmed, the circuitry was tested by placing objects at varying distance in front of the sensor and display distances on a serial LCD.

3.4. Temperature Sensing

In this experiment, the Dallas DS1820 1-Wire™ Digital Thermometer was used to aid the robot in sensing the temperature of its surrounding. The DS1820 uses a unique 1-Wire™ interface that requires only one port pin for communication with the microcontroller. The center pin (DQ) of DS1820 1-Wire™ Digital Thermometer is connected to PORTA.1. For details on the DS1820 temperature sensor, refer to the DS1820 datasheet (DS18S20, 2008) [9].

The PicBasic PRO OWIN and OWOUT commands are used for communicating with DS1820. After the 1-Wire™ temperature sensing device is identified and the unique 64-bit ROM number contained within the device is recorded, communication with the temperature sensor begins. The unique 64-bit ROM code identifies, i) 8-Bit Family Code, ii) Unique 48-Bit Serial Number, and iii) 8-Bit CRC Code. The temperature data is stored in first two data-bytes of the internal scratch pad memory. For details on using 1-Wire™ devices and 1-Wire™ technology refer to (AppNotes1796, 2003). The schematic of DS1820 temperature sensor interfaced to PIC microcontroller is shown in Figure 8. Once, the temperature data is available it is transmitted to a remote PC using IR wireless. The IR wireless serial communication system is established using IR LEDs and an IR detector. These components are similar to the one used for IR obstacle detection and avoidance circuit discussed earlier. The IR wireless communication system is discussed next.
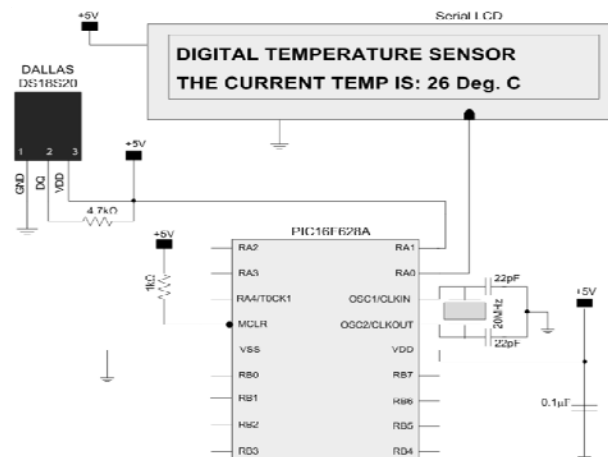

Figure 8. Temperature Sensing Interface

3.5. IR Wireless Communication

A simple IR transmitter requires an IR LED and a means of modulating (turning on and off) the IR LED at carrier frequency. The internal hardware PWM feature of the PIC16F628A microcontroller available through pin PORTB.3 is used to generate the 40kHz carrier for modulating serial data. The IR transmitter circuit is shown in Figure 9.
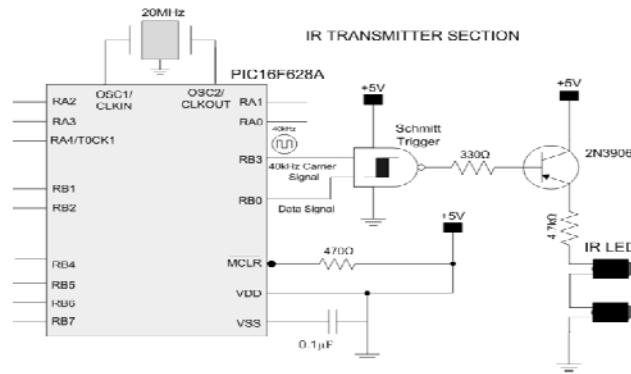
Figure 9. IR Transmitter Interface

In order to configure the PIC16F628A microcontroller for hardware PWM, internal 8-bit registers CCPR1L and CCP1CON are used (Microchip, 2008). The configuration register CCP1CON controls the operation of CCP1 (PORTB.3) pin. Writing a value of 12 to CCP1CON set the Mode Select bits to use and turn PWM output signal on. The hardware PWM can be turned off by writing a value of 0 to CCP1CON. The PWM duty-cycle is set by writing to the CCPR1L register (Microchip, 2008). The hardware PWM is configured in the beginning of the code. The PWM feature is only used to generate the carrier frequency while driving the infrared LEDs during the data transmission. Hardware PWM consumes a lot of power, and when not in use, it is simply turned off.

The transmitter circuit consists of the CCP1 (PORTB.3) pin of the PIC microcontroller connected to one of the two inputs of a 74HCT132 Schmitt Trigger NAND gate. The PORTB.1 pin is used to send serial data stream to the other input of the NAND gate. The Schmitt Trigger NAND gate combines the two incoming signals into an output data signal modulated at 40kHz. The Schmitt Trigger output will only go to logic 0 when both inputs 1 and 2 are at logic 1. This effectively holds the PNP drive transistor off until both inputs are at logic 1. Since the carrier frequency is 40kHz, and considerably faster than serial data, the serial data will cause the PNP transistor to turn on during each logic 1 data-bit. Basically, the serial data turns the carrier on/off for the same bit-time as each data bit in the serial data-stream.

The heart of the receiver circuit shown in Figure 10 is the Vishay Telefunken 40kHz IR detector module. This enhanced data-rate IR detector module can be used for applications operating at 40kHz and baud rates up to 4000bps. It includes a visible light cutoff region to help eliminate interference from visible light source (AppNotes82606, 2008) [10].
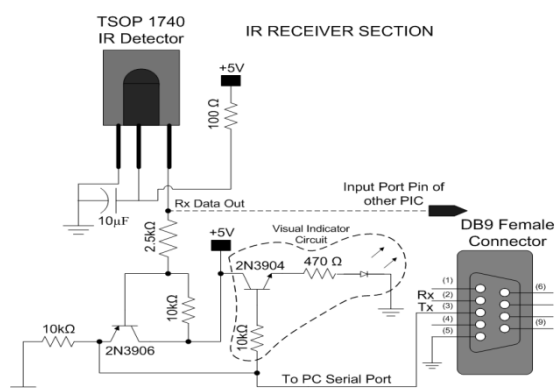

Figure 10. IR Receiver Circuit and Serial Interface

The idle (not receiving data) output state of the infrared detector module is logic 1 or +5V when no infrared energy modulated at 40kHz is detected, and logic 0 or ground when IR energy modulated at 40kHz is striking the face of the detector. The data exiting the PNP transistor in the receiver circuit is inverted. This is important since the default mode for most serial LCD displays, and all PC serial ports is inverted. The

output of the IR detector module can be connected to any available I/O pin of the PIC microcontroller on another robot or to a PC serial port as done in this project. An extra circuit using an NPN transistor and a regular LED is added to indicate data reception. The maximum transmission distance (D) for the IR wireless system depends on two factors; the radiant intensity of emitter and sensitivity of the receiver. The transmission range can be calculated as $D = SQRT(I/E)$, where I is the radiant intensity of the transmitter and E is the sensitivity of the receiver. For the components used, $I = 50mW/sr$ and $E = 0.4mW/m^2$ the system can communicate over a distance of approximately 11m. The maximum peak current (at 100% duty cycle) through the IR LEDs was computed to be approximately 7.04mA. With these parameters, the system was tested at a baud rate of 2400bps over a transmission distance of six meters. This system is capable of achieving baud rates up to 4000bps and transmission distances more than ten meters.

4. IMPLEMENTATION ISSUES AND TROUBLESHOOTING

A few problems were encountered during implementation. Some of these issues were related to servo motors, IR detection, and programming.

4.1. Servo Motor Trouble Shooting

During experiment, servo motors did not function with the standard pulse widths of 1ms, 1.5ms, and 2ms. The exact servo direction constants (reported in section 2) were obtained through manual control of servos at different direction constants. Once the exact directions constants were identified for both the servo motors the problem was resolved. Also it was necessary to provide at least 200mA current for servos to function. The current drive is guaranteed by the use of four AA batteries in the final implementation, but during experiment dual power supply in the laboratory was used to test servos. In addition, while testing for different movements of robots (forward, reverse, left, and right) incorrect servo direction constants were being sent, due to which the robot was not moving in the intended direction. For example, the way left and right servos are placed, for a forward movement, if one servo moves CW the other must move CCW.

4.2. IR Detection Inconsistencies

During the testing of IR obstacle detection, the circuit was not working consistently. It was later realized that the placement of IR LEDs and IR detectors were not placed within the beam deflection angle. This problem was fixed by placing IR LED and corresponding IR detector at a 40 degree cone angle. The placement of heat shrinks on the IR LEDs was also corrected so the IR energy is focused to the front rather then dissipating from sides. After proper placement of IR components, the final robot required only one IR detector for both left and right obstacle detection.

4.3. Programming Editing

While testing the first program, it was found that 20MHz crystal oscillator was not generating any clock signal when logic probe was placed on its pins. The crystal was replaced with another 20MHz crystal, but the problem persisted. Then it was replaced by a 4MHz crystal from another working circuit. The logic probe detected clock pulses on the pins of 4MHz crystal. It was later realized that during the programming of hex code into the PIC microcontroller, the default setting of meLabs programmer was not changed to accommodate 20MHz crystal. By default, the programmer software assumes XT setting, which is for crystal frequency up to 4MHz. For a higher speed crystal it is to be changed to HS_OSC to indicate the use of 20MHz crystal.

5. CONCLUSION

An autonomous mobile robot can be used for a variety of applications, such as search and rescue, fire-fighting, and defusing explosive. When conditions are too hazardous for human beings to operate in, the mobile robots are the ideal solution for completing the task. One mobile robot was designed and implemented as a part of senior design course. A modular approach was used in implementing this robot. The experimental modules discussed in this paper were, servo motor control, IR obstacle detection and avoidance, temperature sensing, and IR wireless communication. Through these experiments, students progressively added functionality to the design. In this process they gained knowledge of circuit design, interfacing sensors, programming microcontrollers, and motor control. This project provided a platform for undergraduate students to apply their knowledge of electronics, sensors, device interfacing and programming to develop a real-world application. The final mobile robot implementation was tested in an application requiring remote temperature sensing. The mobile robot is capable of communicating with another robot or with a host PC at 4000bps and communication range of over 10m.

ACKNOWLEDGEMENTS

REFERENCES

[1] ABET. Criteria for Accrediting Engineering Programs. *Accreditation Board for Engineering and Technology*. Baltimore. 2002.
[2] Bostelman RV, Hong TH, Madhavan R. *Towards AGV Safety and Navigation Advancement - Obstacle Detection using a TOF Range Camera*. Proc. of International Conference on Advanced Robotics (ICAR). Seattle, WA, USA. 2005.
[3] Groer A. After the bomb-defusing robots, here are the robot firefighters. http://www.htlounge.net/categories/Other-Tech/. 2008.
[4] Microchip Technology Inc. 2355 West Chandler Blvd, Chandler, AZ 85224-6199. 2008.
[5] microEngineering Labs, Inc. Box 60039, Colorado Springs, CO 80960. 2008.
[6] Chang T, Hong TH, Legowik S, Abrams M. *Concealment and Obstacle Detection for Autonomouos Driving*. Proc. of the Intl. Association of Science and Technology for Development - Robotics and Application. 1999.
[7] Rawat K, Spruill J, Beasley M, Jones T. *Infrared (IR) Wireless Communication for Application in Mobile Robots*. Proc. of IEEE Southeast Conference. 2007.
[8] Zhao FJ, Guo HJ, Abe KI. A mobile robot localization using ultrasonic sensors in indoor environment," in: *Proc. of the 6th IEEE International Workshop on Robot and Human Communication*. 1997: 52-57.
[9] DS18S20. High Precision One Wire Digital Thermometer, Maxim Dallas Semiconductor, http://datasheets.maxim-ic.com/en/ds/DS18S20.pdf. 2008.
[10] Application Notes 1796. Overview of 1-Wire Technology and Its Use, Maxim Dallas Semiconductor, http://www/maxim-ic.com/an1796.
[11] Application Notes 82606. 2008. Vishay Semiconductors. http://www.mouser.com/catalog/specsheet/vs_82606.pdf.

BIOGRAPHIES OF AUTHORS

Dr. Kuldeep S. Rawat is an Associate Professor and the Head of the Department of Technology at ECSU. He holds a PhD in Computer Engineering from the Center for Advanced Computer Studies at University of Louisiana-Lafayette. Prior to joining UL Lafayette, Dr. Rawat worked as a Research Associate in the Biomedical Engineering Division of Procter & Gamble, India where he developed telemetric data acquisition systems and software applications used in clinical trials. His areas of interest are in designing electronic control applications, signal and image processing, PC-based data acquisition systems, Virtual Instrumentation (VI) for health applications, and instructional technologies. Dr. Rawat is one of the 10 recipients of 2009 HP Innovations in Education grant, which is provided to transform teaching and improve learning in the classroom through innovative uses of technology.

G. H. Massiha, Ph.D., is a Louisiana Board of Region Professor of Engineering. His areas of research interest are robotics, alternative energy, and automation manufacturing.
Dr. Massiha has twenty years of full-time faculty teaching and twenty five years of research experience. His areas of expertise are in experimental and theoretical microelectronics, condensed matter physics, and solar energy research, with emphasis on the engineering reliability and characterization of integrated circuits. In addition, he works on a variety of control and robotics and energy management projects